Multimodal Stereoscopic Movie Summarization Conforming to Narrative Characteristics

Ioannis Mademlis†, Anastasios Tefas†, Nikos Nikolaidis† and Ioannis Pitas†⋆
†Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece
⋆Department of Electrical and Electronic Engineering, University of Bristol, UK

***Abstract*—Video summarization is a timely and rapidly developing research field with broad commercial interest, due to the increasing availability of massive video data. Relevant algorithms face the challenge of needing to achieve a careful balance between summary compactness, enjoyability and content coverage. The specific case of stereoscopic 3D theatrical films has become more important over the past years, but not received corresponding research attention. In the present work, a multi-stage, multimodal summarization process for such stereoscopic movies is proposed, that is able to extract a short, representative video skim conforming to narrative characteristics from a 3D film. At the initial stage, a novel, low-level video frame description method is introduced (Frame Moments Descriptor, or FMoD), that compactly captures informative image statistics from luminance, color, optical flow and stereoscopic disparity video data, both in a global and in a local scale. Thus, scene texture, illumination, motion and geometry properties may succinctly be contained within a single frame feature descriptor, which can subsequently be employed as a building block in any key-frame extraction scheme, e.g., for intra-shot frame clustering. The computed key-frames are then used to construct a movie summary in the form of a video skim, which is post-processed in a manner that also takes into account the audio modality. The next stage of the proposed summarization pipeline essentially performs shot pruning, controlled by a user-provided shot retention parameter, that removes segments from the skim based on the narrative prominence of movie characters in both the visual and the audio modalities. This novel process (Multimodal Shot Pruning, or MSP) is algebraically modelled as a multimodal matrix Column Subset Selection Problem, which is solved using an evolutionary computing approach. Subsequently, disorienting editing effects induced by summarization are dealt with, through manipulation of the video skim. At the last step, the skim is suitably post-processed in order to reduce stereoscopic video defects that may cause visual fatigue.**

***Keywords*—Video Summarization, Stereoscopic Video Description, Column Subset Selection Problem**

I. Introduction

In recent years, the emergence of massive digital video data and their easy global availability, e.g., through popular on-line and mobile Internet channels, has heavily impacted Western societies and accelerated the transformation of their culture into a visual one [1]. This has created a need for succinct and compact presentation of visual digital media. There are several commercial applications where large-scale video footage, possibly available on-line, needs to be analyzed, even on a frame-by-frame basis, demanding human intervention and a great load of effort. Examples include streams from surveillance cameras that may be capturing continuously for many days, videos uploaded to on-line galleries that are available to users for instant browsing, professional capture sessions in the production stage of theatrical films or TV series, where the action described in the script is typically filmed using multiple cameras, or the post-production stage of such material, where its semantic annotation and ordering may be required.

An automated solution is partly offered by *video summarization*, which aims at generating condensed versions of a video stream, through the identification of its most important and pertinent content [2]. The derived video summaries can be subsequently exploited in various applications, like interactive browsing and search systems, thereby offering the user the ability to efficiently view, manage and assess video content [3] [4] [5].

Summarization algorithms initially try to select a set of salient video frames, such as shot key-frames that represent the video content. They vary in type of the performed analysis and / or video summary representation. Moreover, certain techniques are designed to operate on generic video material, whereas others are tailor-made for specific video genres (e.g., sports, news, movies etc.). Besides the video stream, other types of information, such as external information provided by a user, can be exploited in order to create the video summary. Information is extracted by analysing the available modalities (visual, audio or textual) for abstracting intuitive semantics, such as those relevant to objects, events, as well as low-level features from the video stream. The abstracted content that needs to be included in the target summary can be represented as still images (key-frames), a video skim, or by employing graphical and textual descriptions [2]. Due to the inherently subjective nature of the task (there is no such thing as a globally agreed good video summary), the evaluation of a summarization method is typically subjective and qualitative in nature.

Stereoscopic video contains two visual channels and conveys relative-depth information that implicitly ranks each imaged object according to its distance from the camera during video acquisition. Such information is available through *stereoscopic disparity*, an additional image modality that can be extracted by estimating the difference between the two visual channels per video frame. As the availability of stereoscopic 3D video content has increased in recent years, primarily through 3D cinema, the exploitation of disparity-derived relative-depth data to augment summarization performance is a promising

research avenue.

This work presents a complete, multi-stage, multimodal summarization pipeline for 3D movies that exploits audio, visual and stereoscopic disparity information. Initially, the novel Frame Moments Descriptor (FMoD) is introduced, which is a video frame descriptor developed for key-frame extraction, using an intra-shot frame clustering approach. FMoD captures informative image statistics from luminance, color, optical flow or stereoscopic disparity video data, both at global and local scale. Thus, scene texture, illumination, motion and geometry properties may succinctly be contained within a single frame feature descriptor. Depending on the specific problem that needs to be addressed, the descriptor can be computed either in a manner that retains (Frame Moments Descriptor), or discards (Position-Invariant Frame Moments Descriptor) information regarding intra-frame spatial positioning of scene objects.

By filtering out monochrome key-frames, which convey no useful information, and by re-applying clustering on the entire key-frame set, based on a user-provided frame retention percentage parameter, redundant key-frames are discarded. The remaining ones are temporally expanded to key-segments, which are subsequently concatenated, in order to form a stereoscopic video skim. The latter is then post-processed in four ways. First, pre-existing information about temporal speech segments is exploited, in order to expand in time any key-segments that coincide with continuous speech instances. Thus, the audio modality is taken under consideration, with the goal of including in the final skim video highlights featuring complete speech segments, as semantically meaningful movie extracts. Subsequently, the retention percentage parameter is again employed in a proposed Multimodal Shot Pruning (MSP) process, which discards key-segments from the derived video skim, based on which shot they belong to and on pre-existing information about temporal speech (audio) and face (visual) appearance segments. This process is algebraically modeled as a multimodal matrix column subset selection problem, which is solved using an evolutionary computing approach. Thus, a shorter skim is produced in a systematic manner that considers the narrative prominence of movie actors. Next, disorienting editing effects found in the produced skim are tackled, by eliminating temporal jump cuts and removing very short key-segments. Finally, a post-processing step is applied, in order to reduce 3D video quality defects that are an unavoidable by-product of skim-based summarization and may cause visual fatigue. Thus, the source of discomfort during stereoscopic viewing of the final skim is eliminated.

The remainder of this paper is organized in the following way. Section II describes previous work in the field of video summarization. Section III presents in detail the proposed novel summarization method. Section IV describes experiments conducted in order to evaluate the performance of the proposed pipeline in video summarization. In Section V conclusions are drawn from the preceding discussion.

II. Related Work

A. Video Summarization

Generic video summarization algorithms extract key-frame sequences that are presented in temporal order [6] [7]. To achieve this, each video frame is first described by low-level image descriptors, such as global color-based, texture-based or shape-based features [5]. Composite descriptors which may additionally consider visual attention attributes have also been employed [7]. In general, the most commonly employed video frame descriptors are variants of joint image histograms in the HSV color space [8] [9] [10] [11]. Moreover, dimensionality reduction on such color histograms has been attempted, using SVD [12] or PCA [13], in order to decrease the computational cost of the subsequent summarization steps. In a few cases [14] [15], local image region descriptors such as SIFT [16], CSIFT [17] or HOG [18] have been employed for video description, using the Bag-of-Features representation model [19].

In order to extract key-frames, the frame descriptors are typically clustered to create video frame groups, under the assumption that the camera focuses more on important frames [8]. The number of clusters is either set proportionally to the video length [9], or chosen by employing an algorithm, like Furthest-Point-First [10]. After determining the number of clusters, a set of frames that are closest to each cluster center are initially selected as key-frames. Typically, a percentage of the extracted key-frames is filtered out at a refinement post-processing stage and the remaining ones are presented in temporal order to produce a storyboard. This process is improved in [20], by reducing its computational complexity and by adding fuzzy rules based on viewer comments. In [21], a similarity metric is described that assesses the video frame-by-frame, in order to detect whether each video frame should be included in the summary. Frames similar to their previous ones are excluded, while a noise reduction technique based on histograms is applied to exclude homogeneously colored video frames (e.g., black frames). User-defined thresholds can be set to manage the length of the resulting summary.

Approaches other than clustering have been proposed for key-frame extraction. E.g., a computational geometry-based algorithm [**?**] that results in key-frames equidistant to each other in the sense of video content, or a fast method which selects as key-frames the video frames that locally maximize an aggregate intra-frame difference (computed using color features) [22]. However, clustering still dominates the relevant literature due to its simplicity, suitability to the problem and relatively low computational requirements.

Video skims are series of short video segments that are concatenated in the correct temporal order, in order to form a shorter version of the original stream that contains the informative content. Summarization though skimming, instead of still key-frame extraction, allows the inclusion of audio and motion information that can potentially enhance the expressiveness of the video summary [9]. A skim can be derived, in the simplest case, by detecting the silent regions in an audio stream and removing them, therefore significantly decreasing the video length of a full movie [23], or by concatenating video key-segments centered around previously extracted key-frames. Graphical cues can be used to present an additional level of detail to supplement other cues. For example, a two-dimensional color-coded block map of the video stream which distinguishes video segments corresponding to dialogue, explosions and on-screen text, is proposed in [24], where the

end user can manually select the annotated video segments to be included in the summary. A textual cue representation detecting text presence within the video frames is proposed in [25], where the authors detect subtitles within the video stream and check if a sequence of frames belongs to the same dialogue scene. By retaining only the first frame that appears in the same conversation as a key-frame, they can use the key-frames to index the dialogue scenes.

Video content selection and video skimming can also be used in movie post-production. Usually, long videos containing multiple shots are temporally segmented into shots, either manually or automatically. A user attention model is proposed in [26], where visual, audio and textual features are extracted by applying multimodal analysis. A saliency score is computed for each video frame and the most salient frames are selected to be the key-frames. Video key-segments are defined around each key-frame and are concatenated using a fade-in / fade-out approach, in order to form the summarized video skim. A different approach is proposed in [27]. The video stream is first segmented into shots. Then, face detection and tracking [28] [29] [30] is performed on the segmented video clips. Clustering is applied on the extracted facial images, in order to determine which images belong to the same character. The extracted characters are selected to form a character community network, which forms a graph of interactions between the movie characters. Interactions are related to specific video segments, where an importance measure of each interaction is calculated. Redundant interactions are excluded from the video skim, while retaining only video segments that contain interactions with the main characters.

The above described approaches can be applied to generic video content, while specialized methods have also been proposed for specific video genres. E.g., in surveillance videos, motion detection is employed, in order to create summaries that contain sets of human actions, like pedestrian walking. Detected actions taking place at different directions and speed, are fused in a single scene to form a short length video or graphical cue containing as many actions as possible [31] [32]. In another variant, image registration and spatiotemporal motion modelling are employed in videos depicting human actions, in order to summarize them with a single artificial image which is representative of an entire video sequence and expresses a still representation of the dominant motion [33].

B. Disparity Estimation and Stereoscopic Video Summarization

In stereoscopic 3D video content derived from filming with stereo camera rigs (matched pairs of cameras), two images of the scene are available for each video frame, taken at the same time from slightly different positions in world space. From every such *stereo-pair*, composed of a left and a right video frame, a *dense disparity map* that assigns a depth-related disparity value to each image pixel can be estimated from detected pixel correspondences between the two channels [34]. Two different disparity maps can be extracted from a single stereo-pair, associated with the left/right image channel, respectively. When using a parallel camera setup, for each left/right-channel image point $[x, y]^\top$, in pixel coordinates, the corresponding horizontal disparity values are $d^l_{x,y} \leq 0$ and $d^r_{x,y} = -d^l_{x,y}$, while vertical disparities are zero. The closer an imaged object lies to the cameras during image acquisition, the larger is its disparity in absolute value. In contrast, objects considered to be lying at infinity, i.e., positioned very far from the cameras, are projected on pixels with near-zero disparity. When viewed in the theater space during video display, such objects appear in front of the display screen or, in the case of objects at infinity, on the display screen itself.

Disparity estimation, or "stereo matching", has been thoroughly investigated over the past three decades [34]. In the context of this work, due to the massive amount of video frames needed to be processed for the evaluation of the proposed method, an implementation of the SGBM algorithm [35], contained in the publicly available OpenCV library [36], was employed. It provides reasonably accurate disparity estimations in almost real-time. However, they are occasionally noisy and suffer from "blank" pixels, at image regions no correspondence between channels has been detected.

Despite the increased availability of 3D video content [37], a very limited number of video summarization methods operating on stereoscopic or multi-view videos have been presented, mainly using a shot clustering approach. Specifically, an algorithm for multi-view video summarization was proposed in [38], which represents the multi-view video structure by using a spatio-temporal shot graph, clusters shots using random walks and generates the final summary by multi-objective optimization. A semantic content-based approach to stereoscopic video summarization was presented in [39] [40], which performs object segmentation separately on the color and on the disparity channel, then fusing the produced segment maps to extract precise object boundaries. Subsequently, object feature vectors are constructed using multi-dimensional fuzzy classification of segment attributes, including size, location, color and relative-depth, thus allowing K-Means clustering of shots based on the objects they contain. Finally, representative shots are selected from each shot cluster and their key-frames are extracted by minimizing a intra-shot cross-correlation criterion. The same approach was applied in [41] [42] for monocular videos, exploiting motion vectors instead of disparity maps. In [43], stereoscopic video shots are represented using concatenations of various low-level feature descriptors, computed over both the color and the disparity channels, then clustered through a Self-Organizing Map.

III. Multimodal Summarization For Stereoscopic Movies

In this Section, the various stages of the proposed movie summarization pipeline are being presented in detail. Below, the terms "movie" and "video", as well as the terms "actor" and "character" and the terms "summary" and "skim", are used interchangeably.

A. Statistical Stereoscopic Video Description for Key-Frame Extraction

In the proposed approach, the stereoscopic video is assumed to be composed of four temporally ordered sequences of F

frames: V^L, containing luminance frames, V^D containing the corresponding disparity maps, V^C containing the corresponding color frames (the hue channels of the HSV frame representations), and V^O containing the corresponding optical flow maps (e.g., computed for each pair of consecutive luminance video frames, using [44]). Each luminance frame, disparity map, color frame or optical flow map can be considered as a matrix $\mathbf{V}_i^L \in \mathbb{R}^{M\times N}$, $\mathbf{V}_i^D \in \mathbb{R}^{M\times N}$, $\mathbf{V}_i^C \in \mathbb{R}^{M\times N}$, or $\mathbf{V}_i^O \in \mathbb{R}^{M\times N}$, respectively, where $i = 1, \cdots, F$. All available frame sequences are assumed to have been identically partitioned into non-overlapping shots, e.g., by employing the information-theoretic method described in [45].

Key-frames are automatically extracted per shot, by representing each video frame through a feature vector constructed based on the selected image modalities, e.g., luminance and disparity information. The number K of key-frames extracted at each shot is adaptively chosen, i.e., it lies between 2 and a user-provided maximum K_{max}, a parameter that regulates the granularity of the produced summary. Initially, a feature vector is extracted per frame, using a particular feature descriptor. Subsequently, all shot frames are partitioned into K clusters. Finally, the frames closest to the cluster centroids in the feature space, in terms of Euclidean distance, are selected as key-frames. These are subsequently employed to generate a video skim, containing the selected key-frames from all shots and a temporal video segment around each key-frame. The K-Means++ algorithm [46] has proven to be sufficient for shot frame clustering. Other clustering algorithms have been tested and shown to provide similar results.

A novel feature descriptor, called hereafter *Frame Moments Descriptor (FMoD)*, is used for video frame description. It preserves spatial information not available when an entire frame is summarized by a histogram. It can be used with any type of image modality (luminance, color / hue, disparity, optical flow). The Frame Moments Descriptor operates by partitioning a $M \times N$ frame (e.g., $\mathbf{V}_i^L$ or $\mathbf{V}_i^D$) in small blocks of $m \times n$ pixels, where $m < M$ and $n < N$. In each block, two *profile vectors* are computed, one along the horizontal and one along the vertical dimension, by averaging pixel values across block columns / rows, respectively. The result is an n-dimensional and an m-dimensional profile vector. Each of the two vectors is summarized by their first 4 statistical moments (mean, standard deviation, skewness, kurtosis). The resulting 8-dimensional vector $\mathbf{f}_i^v = [m_1^H, m_2^H, m_3^H, m_4^H, m_1^V, m_2^V, m_3^V, m_4^V,]^T$, where v is either L (luminance), D (disparity), C (color / hue) or O (optical flow) compactly captures the statistical properties of the block, as shown in Figure 1. The process is successively repeated d times, for larger values of m and n. In the last iteration, $m = M$ and $n = N$. Finally, all the 8-dimensional vectors are concatenated into a single feature vector that describes the entire frame. This vector set (one per frame) is used for key-frame extraction. The FMoD vector summarizes statistical characteristics of the pixel values (e.g., luminance), in various image regions and in various scales.

FMoD feature extraction was implemented recursively, in a top-down manner, with the image region that is currently being statistically represented at each time, subsequently being recursively partitioned into 4 quadrants. Thus, the total number of 8-dimensional block vectors that are to be concatenated is given by the sum of the first d terms of a geometric progression:

$$1 \cdot 4^0 + 1 \cdot 4^1 + \cdots + 1 \cdot 4^{d-1} = (4^d - 1)/3 \qquad (1)$$

Therefore, the final FMoD feature vector has $8 \cdot (4^d - 1)/3$ dimensions. It compactly describes the video frame in a global and in various local scales, with local information being more spatially focused for higher values of d.

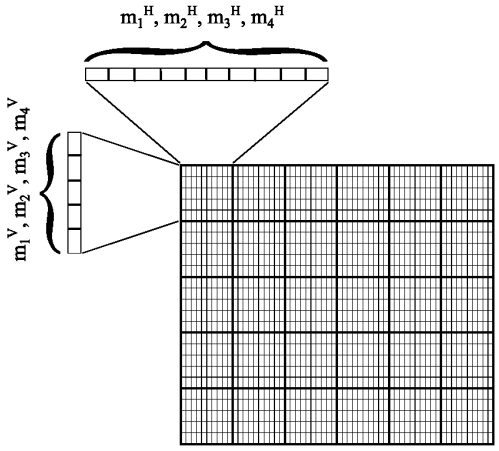


Fig. 1. Statistical summarization of an image block.

Additionally, a variant of FMoD called *Position-Invariant Frame Moments Descriptor (PI-FMoD)*, is proposed. It employs an additional step, in order to discard spatial information from the frame description. This step consists in transforming the set of all 8-dimensional block vectors that compose the FMoD vector into a histogram, using a Bag-of-Features representation [19]. That is, all $(4^d - 1)/3$ block vectors of the frame are clustered into c representative block descriptions, where c is the codebook size parameter. Each block vector is subsequently assigned to the nearest representative block description vector, in terms of Euclidean distance. The number of block vectors assigned to each of the c clusters is an entry in a c-dimensional vector. This vector is followingly transformed into a histogram by L_1-normalization, in order to produce the final c-dimensional frame feature vector.

By employing subjective visual inspection, PI-FMoD frame description was empirically found to perform better than FMoD in the context of key-frame extraction, since spatial information is not necessarily an important factor for the determination of representative shot frames. For instance, a static-camera shot showing an actor walking from the left frame border towards the right one, might be represented by a single key-frame in a satisfactory way. However, in this case, FMoD description would produce significantly different feature vectors for the first and the last shot frame, leading to the unnecessary extraction of multiple key-frames.

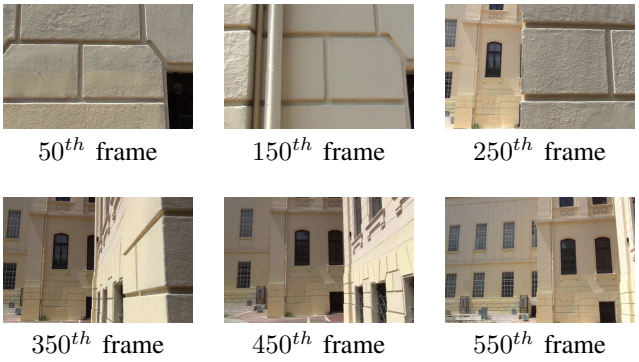


Fig. 2. Example frames from the left color channel of the "Wall" 3D shot.

Whatever the employed descriptor, FMoD or PI-FMoD, one feature vector is computed for the i-th video frame, $i = 1, \cdots, F$, per image modality, e.g., luminance, color / hue, stereoscopic disparity or optical flow. The method used for fusing the various frame feature vectors, e.g., the luminance-derived one and the disparity-derived one, is simple vector concatenation, before performing clustering. Thus, scene texture, illumination and geometry, as well as temporally local motion, can all be taken into account, in order to construct an informative video frame description. Given that the feature vector dimensionality needs to be as low as possible for reducing computational cost, color may be discarded, since it has not been conclusively proven as an important modality for successful summarization [15]. However, a unified, complete frame description would include information derived from all the four image modalities previously considered.

The exploitation of disparity information, and, therefore, scene geometry, potentially leads to the extraction of more representative key-frames, since employing luminance information alone leads to different results than exploiting both disparity and luminance. Figure 2 shows example frames from the "Wall" 3D shot, where the camera pans horizontally from right to left, showing first a wall close-up and subsequently a building in long-shot view. Thus, although the shot frames can be differentiated in terms of disparity, they are mostly homogeneous in luminance and color characteristics, since the wall and the building have similar texture and reflectance properties. Figures 3a,b show two key-frames (K = 2) extracted from the "Wall" when disparity is ignored or is taken into account, respectively. When employing disparity information, two semantically meaningful key-frames can be found, while this is not attained if only the luminance modality is considered.

The number of clusters K, which is equal to the number of extracted key-frames per shot, is determined independently for each shot, by evaluating different clusterings, one for each possible value of K, $K \in \mathbb{N}$, $K \in [2, \cdots, K_{max}]$. Thus, K-Means++ is performed $K_{max} - 1$ times per shot. This adaptive approach does not induce significant computational overhead, since the number of frames per shot is typically less than 100 and clustering is performed very rapidly. The mean silhouette coefficient [47], one of the most simple, robust and well-performing cluster validity indices [48] [49], can be used as the metric for clustering evaluation. The selected value for K is the one corresponding to the clustering with the maximum silhouette.

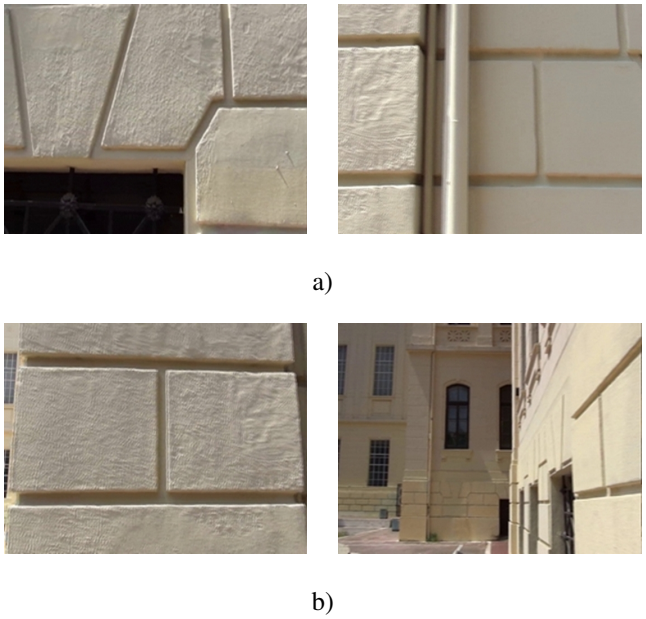


Fig. 3. a) Two left-channel key-frames computed using only luminance information, b) two left-channel key-frames computed by combining luminance and disparity information.

In a post-processing filtering step, monochrome key-frames are discarded from the set of the extracted movie key-frames (based on the variance of their hue component in the HSV color space), which is a common practice in the relevant literature (e.g., in [21]). Subsequently, the remaining video key-frames from all shots are partitioned into $\lfloor Sp \rfloor$ clusters, by reapplying once the K-Means++ algorithm, where S is the total number of extracted key-frames and the user-provided retention parameter p is a percentage that regulates the aggressiveness of frame pruning during this filtering process. The goal is to detect clusters of similar key-frames and remove all frames contained within the same cluster, excluding the one closest to the respective centroid. Thus, a filtered key-frame set of smaller size is derived by considering the entire movie content. This is a late-stage redundancy reduction process, similar to ones typically found in the relevant literature (e.g., in [9]). The greater the value of p, the more key-frames are to be extracted movie-wide and extended into key-segments, thus allowing the proposed summarization method to be adapted to user needs. FMoD, which preserves spatial information, has been found to be a particularly effective descriptor for the detection of multiple similar shot / reverse shot instances, e.g., when two persons are shown alternatingly while they converse, in order to reduce the visual information redundancy inherent in this common film editing technique.

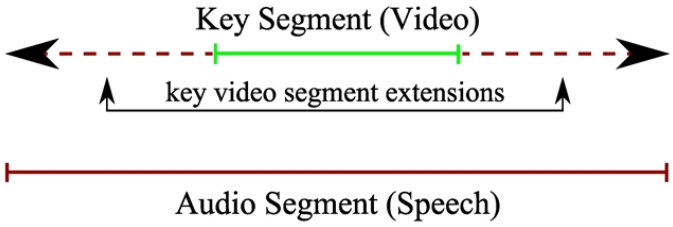


Fig. 4. Audio-assisted extension of a video key-segment.

B. Initial Video Skim Construction

The filtered key-frames are temporally extended, using neighboring video frames, to form key-segments: assuming the i-th video frame is a key-frame, the video segment extending from the $(i - L_{seg})$-th frame up to the $(i + L_{seg})$-th frame is marked as a key-segment. L_{seg} is a user-provided parameter, with the value $L_{seg} = 20$ having been shown to perform well in our experiments. Thus, the initial duration of all key-segments is $D = 2L_{seg} + 1$. Subsequently, each key-segment is confined within the boundaries of its respective shot. Any temporally overlapping key-segments are merged.

The produced key-segments undergo a post-processing step that considers the audio modality: each key-segment is checked against temporally overlapping speech segment appearances, that are derived using speaker diarization algorithms. The latter employ speech segmentation and speaker clustering, in order to temporally segment a video with regard to "who spoke when?", generally without knowing speaker identity [50] [51]. Additional manual annotation may then be applied, in order to assign a label to each speaker, but this is not necessary for this step. Each speech segment consists in a continuous speech instance found in the film. If a video key-segment temporally overlaps with a speech segment, it is suitably being extended to temporally coincide with the latter one. Thus, in the finally produced video skim, no speech instance will be abruptly interrupted and all key-segments containing human voice will feature complete speech instances. This multimodal post-processing stage is depicted in Figure 4. The finally derived key-segments are then concatenated in temporal order to form the video skim.

C. Multimodal Shot Pruning (MSP)

The next post-processing stage in the proposed summarization pipeline performs further key-segment pruning on the initially constructed skim, based on which shot they belong to, since segmentation of the movie into shots is assumed given. This Multimodal Shot Pruning (MSP) process produces a shorter skim which still contains most of the informational content found in the initial one, by "discarding" shots in a systematic manner that considers actor-oriented narrative properties, such as "Who spoke when?" (speakers) and "Who appeared when?" (faces), namely speaker and actor diarization information. As in the case of speakers, each face appearance consists simply of a video segment that starts and ends at the temporal boundaries of an uninterrupted face appearance. Such data may have been acquired through the successive application of face detection [52], face tracking [53], face clustering [54] and label propagation [55] algorithms.

MSP is algebraically modelled as a matrix Column Subset Selection Problem (CSSP) [56], which is briefly discussed here. Assuming a low-rank $M \times N$ matrix $\mathbf{D}$ and a parameter $C < N$, CSSP consists in selecting a subset of exactly C columns of $\mathbf{D}$, which will form a new $M \times C$ matrix $\mathbf{C}$ that captures as much of the information contained in the original matrix as possible. The goal is to construct a matrix $\mathbf{C} \in \mathbb{R}^{M \times C}$ such that the quantity

$$\|\mathbf{D} - (\mathbf{C}\mathbf{C}^{+})\mathbf{D}\|_F \tag{2}$$

is minimized. In the above, $\|\cdot\|_F$ is the Frobenius matrix norm and $\mathbf{C}^{+}$ is the pseudoinverse of $\mathbf{C}$. Thus, the approximation of $\mathbf{D}$ by the smaller matrix $\mathbf{C}$ is expressed in terms of the Frobenius norm in a projection sense: as $\mathbf{D}$ does not have full rank, $\mathbf{C}\mathbf{C}^{+}$ is not simply an identity matrix, but acts as a projection matrix onto the span of the C columns contained in $\mathbf{C}$.

In data analysis, CSSP is an obvious choice for mathematically modelling a feature selection process as an optimization problem. It can be optimally solved by exhaustive search in $\mathcal{O}(N^C)$ time [56], which clearly is a very impractical approach. Thus, approximate algorithms with lower computational complexity have been presented in the relevant literature, with the goal of finding a suboptimal but acceptable solution.

In [57], a metaheuristic approach based on a genetic algorithm is successfully employed for the approximate solution of the CSSP, by directly using Equation (2) as a fitness function. The method is evaluated on several small, randomly generated matrices and is shown to produce good results for a fixed small value of C. In this work, the same metaheuristic approach was adopted and adapted into the proposed pipeline, so that MSP could be modelled and solved as a CSSP.

Specifically, two low-rank, sparse, binary matrices are constructed: $\mathbf{S}, \mathbf{F} \in \mathbb{R}^{V \times S}$, where S is the total number of movie shots and V is the total number of visible speakers, i.e., it is the cardinality of the intersection of the set of all visible faces and the set of all speakers. Typically, $S >> V$. Since temporal speech segment and face appearances are assumed given, $\mathbf{S}$ and $\mathbf{F}$, also referred to as *shot matrices* hereafter, are being filled with binary values:

$$\mathbf{S}_{ij} = \begin{cases} 1, & \text{if the } i\text{-th actor speaks in the } j\text{-th shot,} \\ 0, & \text{otherwise} \end{cases}$$

$$\mathbf{F}_{ij} = \begin{cases} 1, & \text{if the } i\text{-th actor appears in the } j\text{-th shot,} \\ 0, & \text{otherwise.} \end{cases}$$

where $1 \leq i \leq V, 1 \leq j \leq S$.

Subsequently, $\mathbf{S}$ and $\mathbf{F}$ are modified in a Gaussian expansion process, so that each speech / face appearance is "extended" to neighbouring shots. Thus, the initially binary shot matrices are converted to real ones, in a manner that preserves relevant information. That is, for each $\mathbf{S}_{ij} = 1$ / $\mathbf{F}_{ij} = 1$, a discrete approximation of a Gaussian distribution, having its peak at $\mathbf{S}_{ij}$ / $\mathbf{F}_{ij}$, is locally assigned to the entries of the i-th row around

$\mathbf{S}_{ij}$ / $\mathbf{F}_{ij}$, respectively. The standard deviation of the selected probability mass function is chosen so that each appearance is extended only to the dv shots immediately preceding it and following it. Subsequently, shot matrix values derived from different speech / face appearances and corresponding to the same shot matrix entry are added, thus enabling a diffusion of neighboring speech / face appearances. dv may take any value that is less than half the average scene duration (in shots).

This Gaussian expansion process allows a rudimentary form of scene modeling, based on the provided shot segmentation and actor-oriented narrative information. It was included in order to implicitly assist the discrimination between more and less narratively prominent actors. That is, the basis vector sets of the initial shot matrices most likely coincide with the standard basis, with one basis vector corresponding to each visible speaker. However, after the Gaussian expansion, the basis vector sets of the final shot matrices most likely include basis vectors corresponding to the most prominent visible speakers and basis vectors corresponding to combinations of more and less prominent visible speakers. For instance, if the k-th visible speaker is a supporting actor that speaks / appears in scenes (and, therefore, in neigboring shots) only along with a lead actor, there will be no column vector $\mathbf{c}$ of the shot matrices where $\mathbf{c}_i = 0, i \neq k$ and $\mathbf{c}_k \neq 0$.

After the final $\mathbf{S}$ and $\mathbf{F}$ matrices have been constructed, they are implicated in a joint column subset selection problem regulated by a parameter $C = \lfloor S\frac{p}{2} \rfloor$. As previously, S is the total number of movie shots and the user-provided retention parameter p regulates the aggressiveness of shot elimination during this stage. By solving this problem, only key-segments belonging to an optimal subset of shots will be selected to appear in the final video skim, with subset optimality expressed in terms of discarding shots which correspond to shot matrix columns that are linear combinations of other columns. Thus, it is more likely to retain shots where lead actors, or combinations of supporting and lead actors, are present, rather than supporting actors alone.

The desired solution is a set of matrix column indices with cardinality equal to C. Since $\mathbf{S}, \mathbf{F} \in \mathbb{R}^{V \times S}$, for the k-th such index with an assigned value g_k the following hold:

$$k \in \mathbb{N}, \quad k \in [1, \cdots, C]. \tag{3}$$

$$g_k \in \mathbb{N}, \quad g_k \in [1, \cdots, S]. \tag{4}$$

A genetic algorithm is employed to approximate an optimal solution and, as in [57], each candidate is encoded in the form of a sequence of column indices sorted in increasing order. Every such chromosome is of length C. Roulette selection at each iteration is adopted as the mating pool formation strategy. Assuming f_l is the evaluated fitness of the l-th candidate in the current population, this method assigns a selection probability $p^l_{sel} = f_l / \sum_{m=1}^{N} f_m$ to the l-th chromosome.

An order-preserving variant of 1-point crossover [57] is utilized as the main genetic operator. Specifically, in order to combine parent chromosomes $\mathbf{c}^l$ and $\mathbf{c}^m$, a random position k is selected as crossover point and is inspected for suitability. k is considered to be suitable as a crossover point, if the following condition holds:

$$(\mathbf{c}^l_k < \mathbf{c}^m_{k+1}) \wedge (\mathbf{c}^m_k < \mathbf{c}^l_{k+1}). \tag{5}$$

In case Equation (5) does not hold for position k, a different position is selected and inspected. This process continues until either a suitable crossover point has been detected, or all possible positions have been deemed unsuitable. In the former case, crossover is applied and the two parent chromosomes are replaced by their offspring. In the latter case, each of the implicated chromosomes is passed unaltered to the population of the next generation with probability p^l_{sel} or p^m_{sel}, respectively. If $\mathbf{c}^l$ or $\mathbf{c}^m$ is not being retained, it is replaced in the next generation by a copy of the fittest current candidate $\mathbf{c}^n$ with probability p^n_{sel}. If $\mathbf{c}^n$ is also not selected for retention, the process continues with the second fittest of the current candidates, and so on, until a chromosome has been selected.

An order-preserving variant of mutation [57] is employed as the second genetic operator. Specifically, the k-th gene of a chromosome $\mathbf{c}^n$, with an assigned valued of $\mathbf{c}^n_k$, is randomly selected and replaced by a value determined by the neighbouring genes, according to Equation (6):

$$\mathbf{c}^n_k = \begin{cases} rand(0, \mathbf{c}^n_{k+1}), & \text{if} \quad k = 1 \\ rand(\mathbf{c}^n_{k-1}, \mathbf{c}^n_{k+1}), & \text{if} \quad k \in (1, C) \\ rand(\mathbf{c}^n_{k-1}, S+1), & \text{if} \quad k = C. \end{cases} \tag{6}$$

where $rand(a, b)$ uniformly selects a random integer from the interval (a, b). Although this operator ensures a proper ordering of the indices, it has no effect when $\mathbf{c}^n_{k-1}$, $\mathbf{c}^n_k$ and $\mathbf{c}^n_{k+1}$ are successive integers.

The employed fitness function is derived from Equation (2), which is applied to both matrices $\mathbf{S}$ and $\mathbf{F}$. The matrix column indices encoded in the chromosome $\mathbf{c}^n$ which is under evaluation, give rise to the matrices $\mathbf{C}^S$ and $\mathbf{C}^F$, respectively. The former contains a subset of the columns of $\mathbf{S}$, while the latter contains a subset of the columns of $\mathbf{F}$. Thus, the fitness function that needs to be maximized can be expressed as:

$$fit(\mathbf{c}^n) = \frac{1}{\|\mathbf{S} - (\mathbf{C}^S\mathbf{C}^{S+})\mathbf{S}\|_F + \|\mathbf{F} - (\mathbf{C}^F\mathbf{C}^{F+})\mathbf{F}\|_F}. \tag{7}$$

Once the described genetic algorithm has converged to a solution $\mathbf{c}^{best}$, all key-segments belonging to shots not encoded (by their corresponding column index) in $\mathbf{c}^{best}$ are removed from the produced movie skim.

D. Elimination of Disorienting Editing Effects

As a further refinement of the proposed summarization pipeline, an additional key-segment filtering mechanism is applied at this stage. Any key-segments contained within the same shot and separated by less than a second of video duration (e.g., 25 frames in PAL videos), are merged. The purpose is to eliminate abrupt temporal jump cuts in the produced movie skim, i.e. disorienting editing effects that may cause discomfort to the viewer [58]. For similar reasons, any remaining key-segments with too short a duration are also detected and removed, since they have also been empirically found to cause discomfort. A threshold of one second was selected in the context of this work.

E. Elimination of Stereoscopic Video Defects

Given a set of stereoscopic key-segments, annoying *depth jump cuts* may occur at key-segment temporal concatenation points, due to disparity mismatches among consecutive video frames [59]. Such mismatches indicate severe differences in frame depth characteristics, able to cause a disturbing loss of 3D perception during stereoscopic viewing, until the human visual system adapts and the left and right visual channels are fused together to produce a proper 3D scene perception again. Depth jump cuts may be absent in the original movie, due to careful, stereography-aware video editing, but the summarization process unavoidably replaces the original shot transitions with key-segment concatenation points, without taking depth jump cuts into account. Thus, the produced skim may suffer from such defects.

In the final stage of the proposed video summarization pipeline, a previously developed depth jump cut detection and characterization algorithm is applied on the produced video skim and a depth continuity characterization is derived per frame [60]. That is, a depth jump cut is either "absent" (A), "mildly uncomfortable" (MU), "uncomfortable" (U), or "highly uncomfortable" (HU). The employed algorithm operates by detecting rapid changes on temporal mean positive and mean negative disparity signals, both derived from the frames of the stereoscopic video under examination.

In case no depth jump cut is present at a key-segment concatenation point, no further processing is needed. Furthermore, if a U or a HU depth jump cut is detected, a luminance fade out / fade in process is applied to the shot cut, in order to eliminate the source of discomfort during stereoscopic viewing of the video skim. In case a MU depth jump cut is present, a less drastic heuristic technique is employed and described below, aiming at minimizing the presence of such defects in the final video skim.

Between two consecutive key-segments $S_i, S_{(i+1)}$ that cause a MU depth jump cut, the last $\lceil D/4 \rceil$ frames of S_i and the first $\lceil D/4 \rceil$ frames of $S_{(i+1)}$ are exhaustively investigated in pairs, in order to estimate the best possible concatenation point. That is, the frame pair where the Euclidean distance between two of the corresponding disparity maps $\mathbf{V}_f^D$ and $\mathbf{V}_t^D$, with $\mathbf{V}_f^D, \mathbf{V}_t^D$ belonging to the aforementioned relevant subsets of key-segments $S_i, S_{(i+1)}$, is minimal.

IV. Evaluation

Although the evaluation of video summarization methods is an inherently subjective process, there has been an attempt for providing a standard relevant dataset, namely VSUMM [9]. It is accompanied by user-annotated ground truth and a specific evaluation metric, in order to facilitate a more objective comparison between summarization algorithms for research purposes. However, the dataset is simplistic in nature and oriented towards small-scale, single-channel videos, most of which are short, animated clips. Therefore, it was not deemed suitable for the evaluation of the proposed approach, which is specifically designed for stereoscopic, live-action feature films.

In order to assess the performance of the proposed stereoscopic movie summarization pipeline, both an objective and a subjective evaluation scheme were employed. They were performed on 3 stereoscopic Hollywood movies released in 2011, hereby named "Movie1", "Movie2" and "Movie3". Disparity estimation had been applied prior to the evaluation, using the publicly available implementation of the SGBM algorithm [35] in the OpenCV library [36].

Video skims derived with a combination of PI-FMoD / FMoD descriptors were compared against skims derived with image histogram descriptors, for various values of the retention parameter p. For each value of p, multiple FMoD-derived and histogram-derived skims were evaluated, by taking into account different combinations of luminance, color, stereoscopic disparity and optical flow modalities. All histograms were computed with 256 bins per modality, while codebook size c was set to $40N_m$ for PI-FMoD, where $N_m \in \mathbb{N}, N_m \in 1, 2, 3, 4$ is the number of employed modalities at each case. Moreover, d was set to 6, in the case of FMoD, and to 5, in the case of PI-FMoD. These parameter values were found to lead to good results without inducing unacceptably high computational cost. Additionally, K_{max} was set to 5, L_{seg} was set to 20 and dv was conservatively set to 4.

The objective metric employed in our evaluation is the mean silhouette coefficient Sil of the clustering that is performed during the post-processing filtering stage. It holds that $Sil \in \mathbb{R}, Sil \in [0, 1]$ and that a higher value suggests a better clustering. Thus, the proposed video descriptor and the commonly employed histogram descriptors are compared with regard to their performance in clustering, instead of directly with regard to their performance in video summarization, in order to bypass the inherent ambiguity and the subjective nature of the summarization problem.

The 3 scores achieved by each video skim and the corresponding video description method (one for each of the 3 movies) were averaged to compute the aggregate results. In the following notation, L suggests the exploitation of the luminance modality during the description process, C the exploitation of the color / hue modality, D the exploitation of the stereoscopic disparity modality, O the exploitation of the optical flow modality, while LD, CDO, LCD, $LCDO$ and LDO refer to the combination of multiple descriptors computed on the corresponding modalities.

A comparison with the second best performing global descriptor (next to the simple hue histogram), according to [15], i.e., the STIMO descriptor [10], was included. This is a 256-bin joint HSV-space histogram, with 16 value levels in H, 4 levels in S and 4 levels in V, while the simple hue histogram employed in [9] is similar to the Histogram-C descriptor.

The results of the objective evaluation are shown in Table I. As it can be seen, the proposed video descriptor outperforms the typically employed histogram-based description method, as well as the STIMO descriptor, and the best results are achieved when most of the available image modalities (luminance, stereoscopic disparity, color) are exploited. The exception is optical flow, encoding local motion patterns, which does not seem to positively contribute to the process. These findings imply that the richer informational content of

TABLE I. A COMPARISON OF THE AGGREGATE MEAN SILHOUETTE COEFFICIENTS ACROSS THE 3 STEREOSCOPIC MOVIES, FOR DIFFERENT VIDEO DESCRIPTION METHODS AND DIFFERENT VALUES OF THE RETENTION PARAMETER p.

Method	0.5	0.6	0.7	0.8
FMoD-L	0.22	0.22	0.20	**0.16**
FMoD-C	0.21	0.20	0.16	0.12
FMoD-LD	0.18	0.18	0.16	0.13
FMoD-LCD	**0.23**	**0.23**	**0.21**	**0.16**
FMoD-LDO	0.15	0.14	0.13	0.10
FMoD-CDO	0.18	0.18	0.16	0.12
FMoD-$LCDO$	0.19	0.18	0.16	0.13
Histogram-L	0.20	0.19	0.17	0.13
Histogram-C	0.13	0.13	0.13	0.10
Histogram-LD	0.15	0.15	0.14	0.12
Histogram-LCD	0.16	0.17	0.15	0.13
Histogram-LDO	0.16	0.16	0.15	0.12
Histogram-CDO	0.12	0.12	0.11	0.10
Histogram-$LCDO$	0.17	0.17	0.16	0.13
STIMO	0.17	0.17	0.15	0.12

FMoD descriptors, in comparison to histograms, facilitates the determination of more compact and well-separated clusters in the higher-dimensionality feature space that is formed by the concatenation of multiple modalities. Additionally, the mean silhouette coefficients suggest a better clustering when less movie-wide clusters are being used (regulated by the value of the retention parameter p), thus resulting in a shorter, and thus arguably more enjoyable, video skim.

In order to validate these findings, the publicly available Middlebury 2014 [61] stereoscopic image dataset, containing 33 stereoscopic image pairs and corresponding disparity maps, was employed. The dataset was partitioned into 3, 5, 7 and 9 clusters, separately with the FMoD, the histogram and the STIMO descriptor. K-Means++ and the modalities of luminance, color and stereoscopic disparity in different combinations were exploited. The outcomes corroborate previously stated results and are shown in Table II.

To assess the performance of the entire proposed algorithmic pipeline in the task of stereoscopic video summarization itself, rather than in video frame clustering, a subjective evaluation scheme was employed, similar to ones commonly found in the relevant literature. 10 subjects (9 naive and 1 expert) were asked to rate each of the final video skims, in relation to the original movies, in two separate ways: in terms of their *informativeness* and in terms of their *enjoyability*. These two rates per video skim were given on a 0% - 100% scale. All subjects, having recently watched the 3 movies, were independently shown the skims in a consecutive manner and in random order. As in [26], the scale was graded the following way: poor (0% - 40%), fair (40% - 60%), good (60% - 75%), very good (75% - 90%) and excellent (90% - 100%).

TABLE II. A COMPARISON OF THE MEAN SILHOUETTE COEFFICIENTS IN THE MIDDLEBURY 2014 STEREOSCOPIC IMAGE DATASET, FOR DIFFERENT VIDEO DESCRIPTION METHODS AND DIFFERENT NUMBER OF CLUSTERS.

Method	3	5	7	9
FMoD-L	0.27	0.26	0.23	0.21
FMoD-C	0.25	0.24	0.24	0.22
FMoD-LD	0.26	0.24	0.22	0.22
FMoD-LCD	**0.29**	**0.27**	**0.25**	**0.23**
Histogram-L	0.23	0.22	0.20	0.19
Histogram-C	0.20	0.20	0.19	0.18
Histogram-LD	0.21	0.20	0.19	0.17
Histogram-LCD	0.25	0.23	0.21	0.18
STIMO	0.24	0.23	0.22	0.19

In the context of this study, informativeness refers to video content coverage achieved by the produced skim, i.e., to what degree the latter is representative of the original video, retains major plot points and successfully demonstrates major role relationships. Enjoyability refers to the aesthetics of the produced video skim, i.e., to what degree it is composed of semantically complete and coherent scenes, without abrupt and unnatural changes, while simultaneously preserving exciting movie segments and not containing unessential or redundant shots / scenes.

Two main skims were evaluated per movie. One (referred to as "FMoD Pipeline") was constructed using FMoD / PI-FMoD frame descriptors and the entire proposed pipeline, while the other (referred to as "Histogram Pipeline") was constructed using histogram frame descriptors and those stages of the described algorithmic pipeline that are commonly employed in the relevant state-of-the-art literature. That is, intra-shot video frame clustering to extract key-frames, movie-wide key-frame clustering to filter out redundant key-frames, monochrome key-frame filtering and temporal extension to key-segments (without taking the audio modality into account). The presence of MSP in the FMoD Pipeline implies that the corresponding skims are shorter in duration than the ones produced by the Histogram Pipeline, allowing us to evaluate the success of the proposed shot pruning scheme. In both cases, p was set to 0.5, precomputed shot cut boundaries were provided and the image modalities used for video frame description were luminance, stereoscopic disparity and color / hue, according to the results of the objective evaluation.

Additionally, the stereoscopy-aware, object segmentation-based video summarization method presented in [39] was also implemented and evaluated on the three films. The only major deviation from the original method in our implementation was that raw disparity values were employed in video frame feature vector construction instead of actual depth values, to avoid the need for camera auto-calibration (depth map reconstruction from disparity maps requires known camera parameters). This approach clusters video shots based not on low-level frame descriptions, but on their detected (through

TABLE III. A COMPARISON OF THE MEAN INFORMATIVENESS SCORES FOR THE THREE FEATURE FILMS USED IN THE EVALUATION PROCESS.

Method	Movie1	Movie2	Movie3
FMoD Pipeline	70%	74%	72%
Histogram Pipeline	83%	82%	81%
[39] Pipeline	75%	77%	76%

TABLE IV. A COMPARISON OF THE MEAN ENJOYABILITY SCORES FOR THE THREE FEATURE FILMS USED IN THE EVALUATION PROCESS.

Method	Movie1	Movie2	Movie3
FMoD Pipeline	72%	73%	71%
Histogram Pipeline	56%	59%	57%
[39] Pipeline	62%	64%	61%

object segmentation) semantic content. Since it produces a set of key-frames instead of a video skim, the last stages of the Histogram Pipeline were also applied after key-frame extraction, so that the method would arrive at a complete skim for each movie.

The results of the subjective evaluation are shown in Tables III and IV. The FMoD Pipeline achieves significantly better enjoyability scores, at the cost of slightly reduced informativeness, which is to be expected since the duration (in total number of frames) of the skims derived through the FMoD Pipeline is roughly half that of the corresponding Histogram Pipeline skims, as it can be seen in Table V. These results suggest that the additional post-processing stages in the complete proposed algorithmic pipeline successfully remove redundant movie segments, eliminate editing defects and lead to a skim composed of more complete and coherent scenes, while preserving (at least to a degree) major role relations and plot points. The [39] Pipeline, which produced skims slightly shorter in duration than the ones derived from the Histogram Pipeline but longer than the FMoD ones, seems to have been graded by participating subjects according to a similar pattern. Moreover, the heavy reliance of the method on the degree of accuracy of the disparity maps, makes it much more susceptible to disparity noise (which affects object segmentation performance) than the proposed pipeline. This, however, implies that better disparity estimation might, in the future, make method [39] more relevant for stereoscopic feature film summarization.

TABLE V. DURATION (IN FRAMES) OF THE VIDEO SKIMS PER MOVIE. THE DURATION OF THE ENTIRE MOVIE IS ALSO PROVIDED.

Method	Movie1	Movie2	Movie3
FMoD Skim	24644	34907	28020
Histogram Skim	54879	67771	76880
Entire Movie	150358	181763	196224

Figure 5 shows the rate of change in the number of retained key-frames / key-segments, as the proposed algorithmic pipeline progresses, per movie. Stage 1 corresponds to initial key-frame extraction, stage 2 to movie-wide key-frame filtering, stage 3 to monochrome key-frame filtering, stage 4 to overlapping key-segments merge, stage 5 to MSP, stage 6 to small key-segments filtering and stage 7 to jump cut elimination. Additionally, Figure 6 shows the rate of change in mean key-segment duration, as the proposed algorithmic pipeline progresses, per movie. As it can be observed, the described method gradually expands the retained key-segments in temporal terms, while at the same time filters out a large number of them. Thus, a smaller set of more representative, complete and coherent (typically, this implies larger in duration in a semantically meaningful manner) key-segments are acquired, leading to the increased enjoyability achieved when using the FMoD Pipeline.

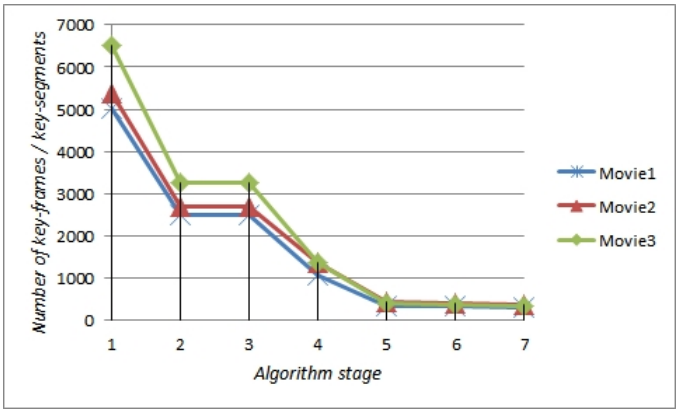


Fig. 5. Reduction in the number of retained key-frames / key-segments, as the proposed algorithmic pipeline progresses, per movie.

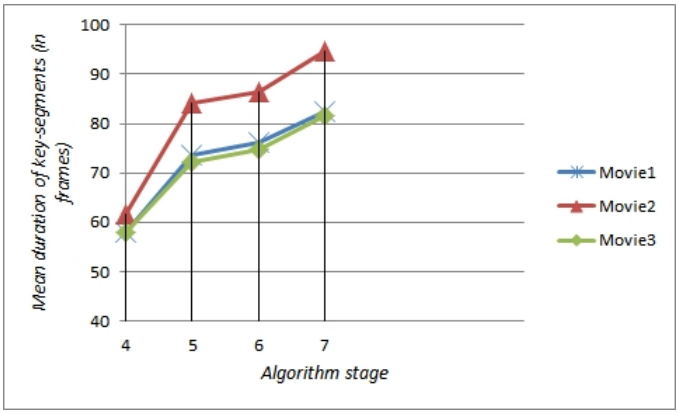


Fig. 6. Increase in the duration of retained key-segments, as the proposed algorithmic pipeline progresses, per movie.

At the MSP stage of the proposed algorithmic pipeline, the following parameters were used for all movies: the maximum number of generations was set to 200, the population size was set to 200, the crossover rate was set to 0.9, the mutation rate was set to 0.005 and the elitism rate was set to 10%. The number of detected visible speakers for Movie1 was 24, for Movie2 was 13 and for Movie3 was 20, while the

corresponding values of C (i.e., the number of movie shots retained after solving the CSSP) are 540 (out of 2161 total detected shots), 511 (out of 2044 shots) and 631 (out of 2524 shots). Figures 7a,b,c show the progression of mean and best population fitness across generations, separately for each movie. As it can be seen, the optimization successfully converges in all cases.

The mean required execution time per video frame across all movies, taking into account all pipeline stages, was 857 milliseconds for the Histogram Pipeline, 1632 milliseconds for the FMoD Pipeline and 1424 milliseconds for the [39] Pipeline. This execution times were measured on a high-end desktop PC, with a Core i7 CPU @ 3.5 GHz and 16 GB RAM. There is an obvious trade-off between summarization quality and execution speed, implying that the proposed method is only suitable for off-line applications.

V. Conclusions

We have proposed a complete, multimodal video summarization algorithmic pipeline for stereoscopic movies, that includes novel video description, shot pruning and post-processing methods. It takes into account visual (shot cut boundaries, scene geometry, texture and illumination), audio (speech instances) and semantic movie characteristics (actor narrative prominence, derived through high-level, semantically meaningful features, such as temporal speech segment and face appearances). To construct the desired movie summary, in the form of a short video skim, the proposed pipeline employs unsupervised learning, algebraic modeling and evolutionary computation techniques, while editing and stereoscopy-related defects are detected and eliminated. The novel video description method (FMoD and PI-FMoD) is favourably compared against the typically used frame histogram descriptions and the competing STIMO description, by employing an objective clustering evaluation metric. The entire proposed pipeline is favourably compared against a typical, clustering-based state-of-the-art summarization pipeline and a competing stereoscopy-aware method, through a standard subjective evaluation process, using three stereoscopic 3D Hollywood movies released in 2011.

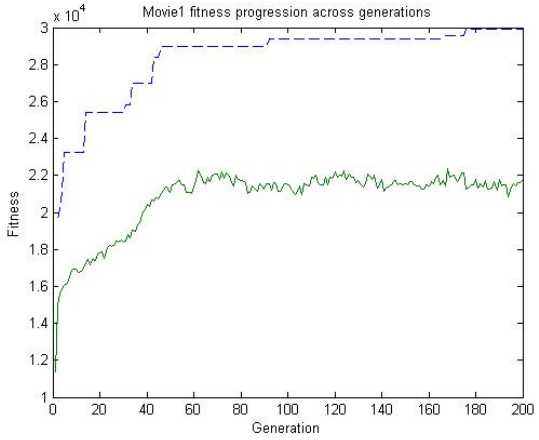
Acknowledgment

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement numbers 287674 (3DTVS) and 316564 (IMPART). This publication reflects only the author's views. The European Union is not liable for any use that may be made of the information contained therein.

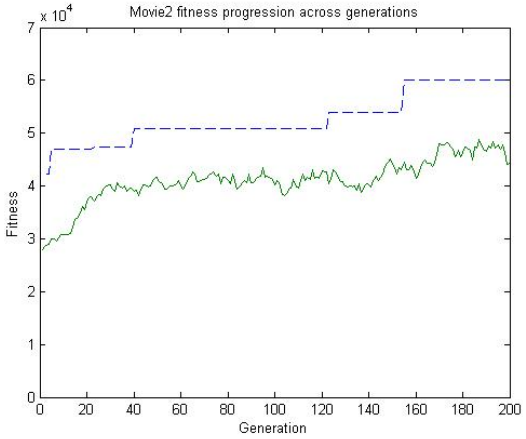
References

[1] M. Sturken and L. Cartwright, *Practices of looking: an introduction to visual culture*, Oxford University Press, 2nd edition, 2009.

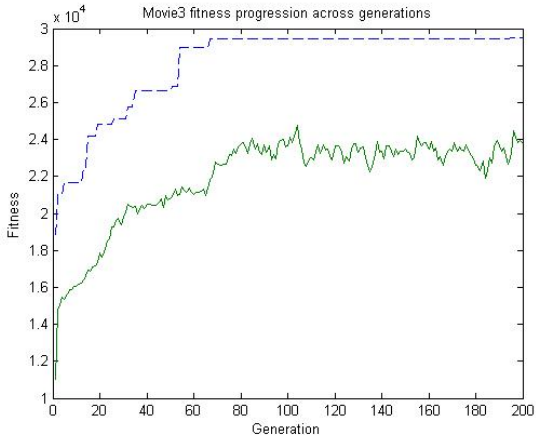
[2] A. G. Money and H. Agius, "Video summarization: A conceptual framework and survey of the state of the art," *Journal of Visual Communication and Representation*, vol. 19, no. 2, pp. 121–143, 2008.



a)



b)



c)

Fig. 7. Mean (solid line) and best (dotted line) population fitness across generations, for a) Movie1, b) Movie2, c) Movie3.

[3] Y. Li, S. H. Lee, C. H. Yeh, and C.-C.J. Kuo, "Techniques for movie content analysis and skimming: tutorial and overview on video abstraction techniques," *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 78–89, 2006.

[4] M. S. Lew, N. Sebe, C. Djeraba, and R. Jain, "Content-based multimedia information retrieval: state of the art and challenges," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 2, no. 1, pp. 1–19, 2006.

[5] W. Hu, N. Xie, L. Li, X. Zeng, and S. Maybank, "A survey on visual content-based video indexing and retrieval," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 41, no. 6, pp. 797–819, 2011.

[6] C. Gianluigi and S. Raimondo, "An innovative algorithm for key frame extraction in video summarization," *Journal of Real-Time Image Processing*, vol. 1, no. 1, pp. 69–88, 2006.

[7] P. Kelm, S. Schmiedeke, and T. Sikora, "Feature-based video key frame extraction for low quality video sequences," *10th Workshop on Image Analysis for Multimedia Interactive Services*, vol. 1, no. 1, pp. 25–28, 2009.

[8] Y. Zhuang, Y. Rui, T.S. Huang, and S. Mehrotra, "Adaptive key frame extraction using unsupervised clustering," *Proceedings of International Conference on Image Processing (ICIP)*, vol. 1, pp. 866–870, 1998.

[9] S. E. F. De Avilla, A. P. B. Lopes, A. L. Jr. Luz, and A. A. Araujo, "VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method," *Pattern Recognition Letters*, vol. 32, no. 1, pp. 56–68, 2011.

[10] M. Furini, F. Geraci, M. Montangero, and M. Pellegrini, "STIMO: Still and moving video storyboard for the web scenario.," *Multimedia Tools and Applications*, vol. 46, no. 1, pp. 47–69, 2010.

[11] P. Mundur, Y. Rao, and Y. Yesha, "Keyframe-based video summarization using delaunay clustering," *International Journal on Digital Libraries*, vol. 6, no. 2, pp. 219–232, 2006.

[12] Yihong Gong and Xin Liu, "Video summarization and retrieval using singular value decomposition," *Multimedia Systems*, vol. 9, no. 2, pp. 157–168, 2003.

[13] T. Wan and Z. Qin, "A new technique for summarizing video sequences through histogram evolution," *Signal Processing and Communications (SPCOM), 2010 International Conference on*, pp. 1–5, 2010.

[14] J. Li, "Video shot segmentation and key frame extraction based on SIFT feature," *International Conference on Image Analysis and Signal Processing (IASP)*, pp. 1–8, 2012.

[15] E J.Y. Cahuina and G. C. Chavez, "A new method for static video summarization using local descriptors and video temporal segmentation," *Graphics, Patterns and Images (SIBGRAPI), 2013 26th SIBGRAPI-Conference on*, pp. 226–233, 2013.

[16] D. G. Lowe, "Object recognition from local scale-invariant features," *Computer Vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2, pp. 1150–1157, 1999.

[17] Koen van de Sande, Theo Gevers, and Cees Snoek, "Evaluating color descriptors for object and scene recognition," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1582–1596, Sept. 2010.

[18] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *International Conference on Computer Vision & Pattern Recognition*, vol. 2, pp. 886–893, June 2005.

[19] G. Csurka, C. Bray, C. Dance, and L. Fan, "Visual categorization with bags of keypoints," *ECCV, Workshop on Statistical Learning in Computer Vision*, 2004.

[20] M. Pournazari, M. Fariborz, and Amir M.E.M., "Video summarization based on a fuzzy based incremental clustering," *International Journal of Electrical and Computer Engineering*, 2014.

[21] J. Almeida, N. J. Leite, and R. dS. Torres, "Vison: Video summarization for online applications," *Pattern Recognition. Letters*, vol. 33, no. 4, pp. 397–409, 2012.

[22] N. Ejaz, T. B. Tariq, and S. W. Baik, "Adaptive key frame extraction for video summarization using an aggregation mechanism," *Journal of Visual Communication and Image Representation*, vol. 23, no. 7, pp. 1031–1040, Oct. 2012.

[23] M. Furini and V. Ghini, "An audiovideo summarisation scheme based on audio and video analysis," *Proceedings of the IEEE Consumer Communications and Networking Conference*, 2006.

[24] R. Lienhart, S. Pfeiffer, and W. Effelsberg, "Video abstracting," *Communications of the ACM*, vol. 40, no. 12, pp. 54–62, 1997.

[25] J. Liu B. Luo, X. Tang and H. Zhang., "Video caption detection and extraction using temporal information," *International Conference on Image Processing*, vol. 1, no. 1, pp. 297–300, 2003.

[26] G. Evangelopoulos, A. Zlatintsi, A. Potamianos, P. Maragos, K. Rapantzikos, G. Skoumas, and Y. Avrithis, "Multimodal saliency and fusion for movie summarization based on aural, visual, and textual attention," *IEEE Transactions on Multimedia*, vol. 15, no. 7, pp. 1553 – 1568, 2013.

[27] C.W. Lin C.M. Tsai, L.W. Kang and W. Lin, "Scene-based movie summarization via role-community networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 11, pp. 1927 – 1940, 2013.

[28] O. Zoidi, N. Nikolaidis, A. Tefas, and I. Pitas, "Stereo object tracking with fusion of texture, color and disparity information," *Signal Processing: Image Communication*, vol. 29, no. 5, pp. 573–589, 2014.

[29] H. Baltzakis, A. Argyros, M. Lourakis, and P. Trahanias, "Tracking of human hands and faces through probabilistic fusion of multiple visual cues," *Computer Vision Systems*, pp. 33–42, 2008.

[30] M. M. Elmansori and K. Omar, "An enhanced face detection method using skin color and back-propagation neural network," *European Journal of Scientific Research*, 2011.

[31] W. Fu, J. Wang, L. Gui, H. Lu, and S. Ma, "Online video synopsis of structured motion," *Neurocomputing*, 2014.

[32] K. Streib and J. Davis, "Summarizing high-level scene behavior," *Machine Vision and Applications*, vol. 25, no. 1, pp. 229–244, 2014.

[33] N. Vasconcelos and A. Lippman, "A spatiotemporal motion model for video summarization," in *CVPR*. 1998, pp. 361–366, IEEE Computer Society.

[34] D. Scharstein and R. Szeleiski, "A taxonomy and evaluation of dense two frame stereo correspondence algorithm," *IEEE International Journal of Computer Vision*, vol. 47, no. 1/2/3, pp. 7–42, 2002.

[35] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *???? Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328–341, 2008.

[36] G. Bradski, A. Kaehler, and V. Pisarevsky, "Learning-based computer vision with Intels open source computer vision library," *Intel Technology Journal*, vol. 9, no. 2, pp. 119–130, 2005.

[37] O. Schreer, P. Kauff, and T. Sikora, *3D videocommunication: Algorithms, concepts and real-time systems in human centred communication*, Wiley, 2006.

[38] Y. Fu, Y. Guo, Y. Zhu, F. Liu, C. Song, and Z.H. Zhou, "Multi-view video summarization," *IEEE Transactions on Multimedia*, vol. 12, no. 7, pp. 717 – 729, 2010.

[39] N. Doulamis, A. Doulamis, Y.S. Avrithis, K.S. Ntalianis, and S.D. Kollias, "Efficient summarization of stereoscopic video sequences," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 4, pp. 501517, 2000.

[40] N. Doulamis, A. Doulamis, Y.S. Avrithis, K.S. Ntalianis, and S.D. Kollias, "An optimal framework for summarization of stereoscopic video sequences," in *PROCEEDINGS OF INTERNATIONAL WORKSHOP ON SYNTHETIC - NATURAL HYBRID CODING AND THREE DIMENSIONAL IMAGING*, 1999.

[41] A. Doulamis, N. Doulamis, and S. Kollias, "A fuzzy video content representation for video summarization and content-based retrieval," *Signal Processing*, vol. 80, no. 6, pp. 1049 1067, 2000.

[42] N. D. Doulamis, A. D. Doulamis, Y. Avrithis, and S. D. Kollias, "A stochastic framework for optimal key frame extraction from mpeg video databases," in *Multimedia Signal Processing, 1999 IEEE 3rd Workshop on*, 1999, pp. 141–146.

[43] K. Papachristou, A. Tefas, N. Nikolaidis, and I. Pitas, "Stereoscopic video shot clustering into semantic concepts based on visual and disparity information," in *Image Processing (ICIP), 2014 IEEE International Conference on*, 2014, pp. 5472–5476.

[44] G. Farneback, "Two-frame motion estimation based on polynomial expansion," in *SCIA*. 2003, vol. 2749 of *Lecture Notes in Computer Science*, pp. 363–370, Springer.

[45] Z. Cernekova, I. Pitas, and C. Nikou, "Information theory-based shot cut/fade detection and video summarization.," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 1, pp. 82–91, 2006.

[46] D. Arthur and S. Vassilvitskii, "K-means++: the advantages of careful seeding," *SODA '07: Proceedings of the 18th annual ACM-SIAM symposium on discrete algorithms*, pp. 1027–1035, 2007.

[47] P. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, no. 1, pp. 53–65, 1987.

[48] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J. M Pérez, and I. Perona, "An extensive comparative study of cluster validity indices," *Pattern Recognition*, vol. 46, no. 1, pp. 243–256, 2013.

[49] L. Vendramin, R. Campello, and E. R. Hruschka, "On the comparison of relative clustering validity criteria," *SDM*, pp. 733–744, 2009.

[50] S. E. Tranter and Douglas A. Reynolds, "An overview of automatic speaker diarization systems," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1557–1565, 2006.

[51] S. Meignier and T. Merlin, "Lium spkdiarization: An open source toolkit for diarization," *CMU SPUD Workshop, Dallas (Texas, USA)*, March 2010.

[52] G. N. Stamou, M. Krinidis, N. Nikolaidis, and I. Pitas, "A monocular system for automatic face detection and tracking," *Proceedings of Visual Communications and Image Processing*, vol. 5960, 2006.

[53] O. Zoidi, A. Tefas, N. Nikolaidis, and I. Pitas, "Stereo object tracking with fusion of texture, color and disparity information," *Signal Processing: Image Communication*, vol. 29, no. 5, pp. 573–589, 2014.

[54] G. Orfanidis, A. Tefas, N. Nikolaidis, and I. Pitas, "Facial image clustering in stereoscopic videos using double spectral analysis," *Signal Processing: Image Communication*, vol. 33, pp. 86–105, 2015.

[55] O. Zoidi, A. Tefas, N. Nikolaidis, and I. Pitas, "Person identity label propagation in stereo videos," *IEEE Transactions on Multimedia*, vol. 16, no. 5, pp. 1358–1368, 2014.

[56] C. Boutsidis, M. W. Mahoney, and P. Drineas, "An improved approximation algorithm for the column subset selection problem," in *Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms*. 2009, SODA '09, pp. 968–977, Society for Industrial and Applied Mathematics.

[57] P. Kromer, J. Platos, and V. Snasel, "Genetic algorithm for the column subset selection problem," *Complex, Intelligent and Software Intensive Systems*, pp. 16–22, 2014.

[58] D. Bordwell and K. Thompson, *Film Art: An Introduction*, McGraw Hill, 8 edition, 2006.

[59] B. Mendiburu, *3D movie making. Stereoscopic digital cinema from script to screen*, Focal Press, 2009.

[60] S. Delis, I. Mademlis, N. Nikolaidis, and I. Pitas, "Automatic detection of 3D quality defects in stereoscopic videos using binocular disparity," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. PP, no. 99, pp. 1–1, 2016.

[61] D. Scharstein, H. Hirschmller, Y. Kitajima, G. Krathwohl, N. Nesic, X. Wang, and P. Westling, "High-resolution stereo datasets with subpixel-accurate ground truth," in *GCPR*. 2014, vol. 8753 of *Lecture Notes in Computer Science*, pp. 31–42, Springer.